

ARG_AI

Infrastructure & Implementation

Industrial ML · Retrieval-Augmented Generation · Private and External LLM

سند مهندسی آماده‌سازی، نصب و راه‌اندازی زیرساخت هوش مصنوعی ارگ برای پروژه‌های صنعتی



Design basis for engineering, plant IT, OT security, data science and operations.

01

ML & Data

02

RAG & Knowledge

03

Local / API LLM

این طرح بر اساس رابط فعلی ARG_Bms و بخش‌های قابل مشاهده Main Backend تهیه شده است. قابلیت‌های ML و RAG در ARG_AI بر اساس اعلام مالک محصول موجودند. این سند روش استقرار صنعتی و اتصال به داده واقعی را طراحی می‌کند؛ جزئیات کد سرور هنگام تحویل بسته اجرایی تطبیق داده می‌شوند.

Document status	Value
Revision	1.1 / Existing ML + RAG deployment basis
Scope	ARG_AI serving, ML pipeline, RAG, local LLM, external providers and operation
Operating principle	Industrial telemetry is published through the approved OT/DMZ path; AI is advisory
Procurement status	Resource estimates require benchmark of selected models, site data and concurrent users

Why AI belongs in the digital twin

همزاد دیجیتال وضعیت دارایی را از داده معتبر بازنمایی می کند؛ اجزای AI هر کدام کار متفاوتی روی همین وضعیت انجام می دهند.

Component	Input in the twin	Output and actual use
Digital twin state	Time-aligned telemetry, asset hierarchy, ENID, constraints and quality flags	Current and historical state of equipment and process
ML	Versioned features from historian observations and maintenance outcomes	Anomaly score, failure likelihood and forecast with uncertainty
RAG	Approved SOPs, manuals and P&IDs with revision and ACL	Relevant permitted source passages and citations
LLM	Question, authorized twin snapshot, ML evidence and retrieved passages	Explanation, comparison and structured advisory summary
Agent	Bounded plan and approved read-only tools over twin, ML and RAG	Multi-step investigation with traceable tool calls and human review

مثال: افزایش لرزش پمپ از historian وارد twin می شود؛ ML نمره ناهنجاری می دهد؛ RAG دستورالعمل معتبر را می یابد؛ LLM یافته ها را توضیح می دهد؛ Agent فقط با مجوز کاربر شواهد لازم را فراخوانی می کند. نتیجه پیشنهاد است و مسیر فرمان به DCS/SIS نیست.

Sources: C01 · C06 · S13 · S14

Industrial use cases and measurable outputs

هر قابلیت باید مسئله عملیاتی، شواهد قابل نمایش و مالک پذیرش مشخص داشته باشد.

Scenario	AI mechanism	Evidence in UI	Owner
Pump anomaly triage	ML detects deviation; Agent reads approved trend and SOP via RAG; LLM explains hypotheses	Tag/time/quality, score/version and cited SOP revision	Reliability
Energy consumption forecast	ML predicts demand; LLM compares forecast with baseline and constraints	Horizon, uncertainty, feature freshness and measured baseline	Energy
Operator document question	RAG filters by role; LLM answers with citations or abstains	Page/section, revision, index time and scope	Operations
What-if assessment	Twin scenario engine evaluates change; Agent gathers inputs; LLM states assumptions	Inputs, baseline, engine version and constraint violations	Process

پیش از آموزش، متغیر هدف و روش برچسب گذاری با مهندس فرایند تعیین می شود. پیش بینی روی داده واقعی سایت ارزیابی و سپس وارد چرخه تصمیم می شود.

Sources: C01 · C02 · C03 · S13

Decision mechanism and trust boundary

این جریان محل شواهد، فراخوانی Agent و تصمیم نهایی انسان را مشخص می کند.

Stage	Input	Output / gate
01 OT publication	Approved historian tags via DMZ	Validated twin snapshot with source time and quality
02 ML inference	Versioned features from snapshot	Score/forecast with model version and uncertainty
03 Knowledge retrieval	Asset ID + authenticated role	Permission-filtered, revisioned passages
04 Agent investigation	User intent + bounded read-only tools	Traceable ML/twin/RAG calls; no controller write
05 LLM explanation	Authorized facts + retrieved evidence	Cited advisory result or abstention
06 Human review	Advisory result + plant procedure	Authorized operator/process engineer decision

Agent ابزارهای مجاز، محدودیت تعداد فراخوانی، زمان اجرا و ثبت کامل trace دارد. ارزیابی prompt injection و کنترل دسترسی اسناد پیش از آزمایش عملیاتی الزامی است.

Sources: C01 · C06 · S12 · S14

Commissioning roadmap: gates before installation

از مسئله عملیاتی شروع می‌کنیم؛ هر مرحله خروجی قابل بازبینی برای مرحله بعد دارد.

Gate	Action	Evidence to advance
G0 Define value	Select 2–3 assets, use cases, KPI, owner and failure consequences	Signed scope and representative questions
G1 Audit ARG_AI	Obtain released binary/source, OpenAPI, DB schema, dependencies and config	Versioned release manifest and interface map
G2 Approve data	Classify telemetry/documents; map OT publication and AI ingress	Data contract, ACL matrix, approved conduit
G3 Qualify capacity	Benchmark ML, embedding and LLM with target context and load	RAM/VRAM, latency, storage and qualified BOM
G4 Install platform	Provision hosts, DNS/TLS, backup, secrets, database and observability	As-built inventory and smoke test
G5 Connect existing ML/RAG	Replace sample UI data; validate features, ingest documents and evaluate retrieval	Source-to-UI trace and permission test
G6 Enable LLM/Agents	Configure provider policy and read-only tools; test cost and prohibited routes	Agent trace, policy and failure drill
G7 FAT/SAT/operate	Run scenario, load, restore and OT-separation tests; train users	Signed results, rollback and handover

صفحات بعد جزئیات فنی، تجهیزات، پیکربندی، آزمون و معیار تحویل هر گام را می‌دهند. خرید GPU پس از بارسنجی مدل انتخابی نهایی می‌شود.

ARG_AI existing systems: implementation baseline

این سند وجود سامانه‌های ML و RAG را بر اساس اعلام مالک محصول مبنا می‌گیرد و مسیر استقرار صنعتی همان سامانه‌ها را تعیین می‌کند.

Current capability	Deployment integration to complete	Acceptance proof
ML training and analysis in ARG_AI	Bind existing training endpoint/jobs to approved historian features, target labels, dataset versions, model artifacts and registry.	Real-data train/analysis trace: source row → feature → training run → versioned model → inference response.
RAG in ARG_AI	Point current ingestion and retrieval to approved document repository; enforce revision, asset identity, tenant/role ACL and citations.	Ingest and query logs, indexed revision, denied-role test and citation to correct source span.
Provider settings and chat UI	Reconcile actual server adapters with local model and external API route; protect secrets and test fallback.	End-to-end request IDs, redacted logs, provider selection and outage response.
Digital-twin / what-if endpoints	Map physical assets and time-aligned tags to current simulation contract.	Approved input snapshot, reproducible scenario and explicit limitations.

وجود ML و RAG بر اساس تأیید مستقیم مالک سیستم است. نوع الگوریتم، محل ذخیره مدل، پیاده‌سازی فعلی RAG و وضعیت اجرای سایت هنوز از کد سرور مستقل بررسی نشده‌اند؛ این قراردادها هنگام تحویل نسخه اجرایی با واقعیت آن تطبیق می‌شوند.

Sources: C01 · C02 · C03

Evidence boundary and design decisions

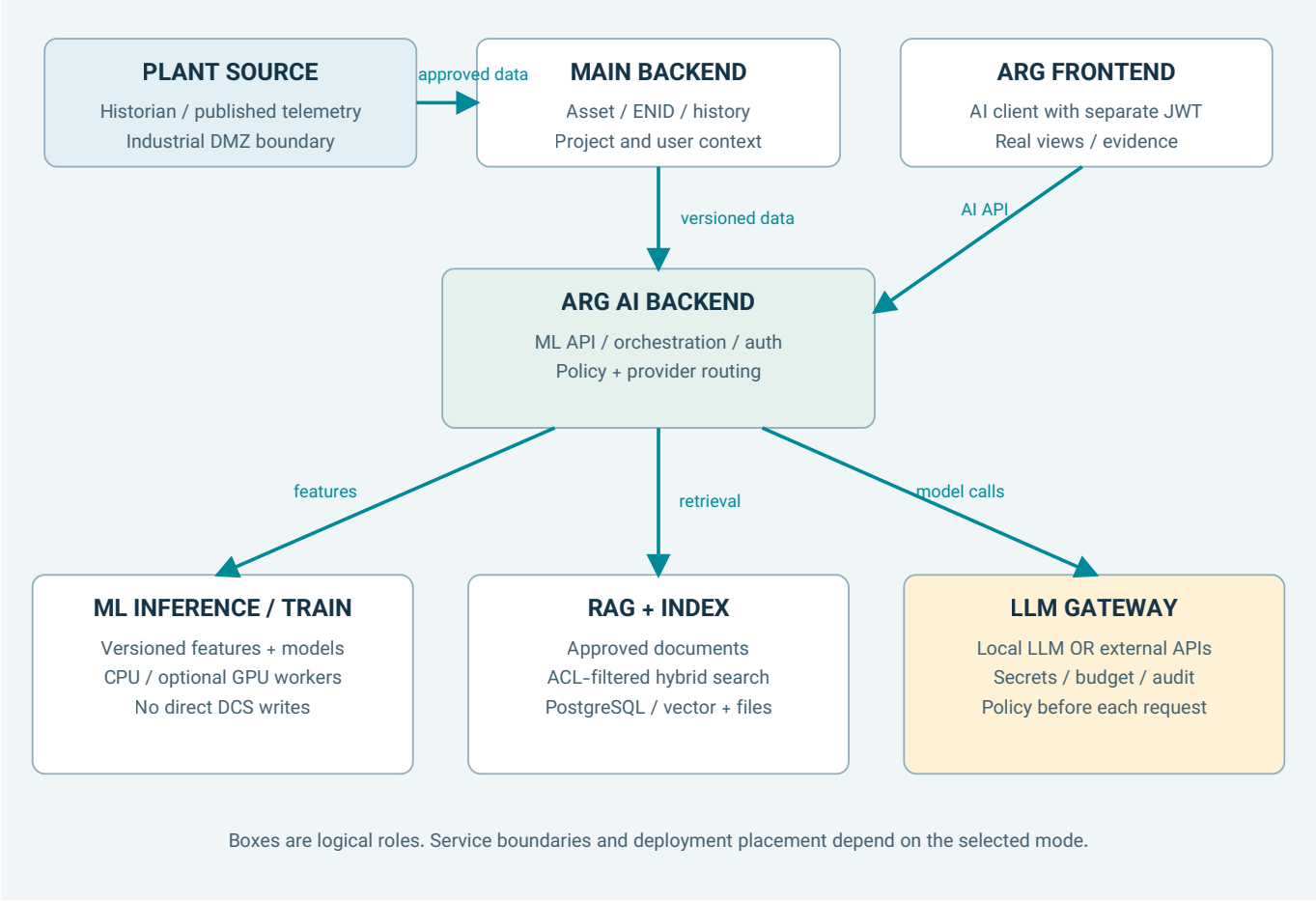
برای جلوگیری از اشتباه معماری، داده‌ای که از کد دیده شده با طراحی پیشنهادی و موارد نیازمند تأیید سرور جدا شده است.

Class	Observed or proposed detail	Use
Client verified	AIAPIService contacts a separate ARG_AI endpoint; maintains an independent JWT; routes include chat, ML, training, provider settings, digital-twin simulate and what-if	Interface and integration acceptance basis
Main Backend verified	Main Backend has its own AI model/embedding clients with different API contracts; action gate is separate	Do not equate its embedded assistant with the independent AI server
Demo input identified	AnomalyDetection and ConsumptionPrediction pages generate sample arrays before invoking ML API	Replace with authorized plant feature pipeline before SAT
Historical host evidence	An earlier inventory recorded ArgAI.jar behind ai.argsrv.com on an internal port; current artifact and datastore are unverified	Orientation only; reconcile actual service and version
Proposed engineering	Industrial deployment of existing ML/RAG, optional local LLM workers, provider gateway, stores and monitoring	Integrate existing ARG_AI ML/RAG; deploy local LLM and gateway roles to match actual server contracts
Site-dependent	Model families, training frequency, document pages, context window, concurrency and availability	Freeze for procurement and FAT

Sources: C01 · C02 · C03 · C04 · C05 · C06

Target architecture

سامانه داده زنده و اسناد صنعتی را از مرزهای مورد تأیید دریافت می‌کند. مسیر تحلیل AI به تجهیزات کنترلی فرمان مستقیم نمی‌دهد.



Plane	Trust and lifecycle rule
Data	Read approved historian/DMZ publications; preserve source time, quality, unit, ENID and revision.
Application	Independent AI API authenticates requests and enforces site/project scope.
Knowledge	Document revisions, permissions, embedding model and index all versioned together.
Inference	LLM gateway chooses on-site or allowed provider per classification and budget.
Feedback	AI returns score/forecast/recommendation with model version and evidence; plant decision remains with authorized operators.

Sources: C01 · C04 · S13 · S14

Zones, conduits and external egress

اتصال اینترنت برای حالت محلی لازم نیست. در حالت API، تنها سرویس واسطی که مجوز دارد از خروجی کنترل شده به ارائه دهنده متصل می شود.

Conduit and initiator	Destination	Transport	Control
D01 Approved collector	Industrial DMZ publication	Site-approved HTTPS/MQTT	Read or publish only; no direct DCS route from AI zone
D02 AI feature worker	Main Backend / historian projection	Private HTTPS / DB read only if authorized	Scoped data contract and rate cap
D03 Browser	AI API via HTTPS proxy	443	Separate AI JWT; same-origin proxy preferred after frontend change
D04 AI orchestrator	RAG DB / document store	Private DB/TLS + file service	Per-tenant credentials; no public DB access
D05 AI orchestrator	Local LLM service	Private TLS	Service authentication; no public model server port
D06 Provider gateway	Allowlisted external APIs	443 via egress proxy	Data policy, DNS allowlist, TLS verification, audit and budget
D07 Management	Model registry / backups / metrics	Site-controlled management network	Administrator roles, immutable logs and restore exercise

دستگاه های OT و مسیرهای SIS مستقل باقی می ماند. ارسال داده خام تولید یا اسناد محرمانه به API بیرونی بدون مجوز طبقه بندی و سیاست سایت در طرح مجاز فرض نشده است.

Sources: S14 · S02 · S08

Workload discovery before sizing

تعداد پارامترهای مدل به تنهایی برای انتخاب GPU کافی نیست. باید شکل درخواست و حجم اسناد اندازه گیری شود.

Input	Measurement needed	Sizing consequence
Chat	Peak concurrent sessions, input/output tokens, prompt length, percentile latency	GPU memory for weights + KV cache; scheduler concurrency
RAG	Documents, pages, images, OCR rate, chunks, vector dimension and reindex window	CPU/RAM ingest; vector and document storage
ML	Feature series/day, sampling, training windows, model families, retrain cadence	CPU/GPU training node, feature store and batch queue
Historian	Assets/ENIDs, time-series retention, timestamp quality, source history API limits	Extractor scheduling and data correctness
Availability	RTO/RPO, offline operation, burst, provider outage	Standby/backup and local fallback decision
Data policy	Public, internal, confidential, OT-sensitive classes	Allowed provider, log redaction and retention

LLM memory lower bound (illustrative): $\text{weights_bytes} \approx \text{parameters} \times \text{bits_per_weight} / 8$.
Total VRAM $\approx \text{weights} + \text{KV cache}(\text{context} \times \text{concurrent sequences} \times \text{model geometry}) + \text{runtime/workspace}$.
RAG raw vector payload $\approx \text{chunk_count} \times \text{dimensions} \times \text{bytes_per_component}$; index + metadata are additional.
Capacity is qualified at target context and concurrency, not from weight size alone.

Sources: S01 · S04 · S05

Deployment modes and purchase gate

سه مسیر، با همان API های ARG_AI قابل طراحی اند؛ قرارداد احراز هویت و پردازش داده برای هر کدام جداگانه آزمون می شود.

Mode	Compute / location	Strength and dependencies	Purchase gate
API-led	ARG_AI + RAG/ML on site; inference from approved external provider	No local LLM GPU required; WAN, quota, policy and API fees apply	Egress permission + data controls + provider SLA
Fully local	ARG_AI + RAG + separate Linux GPU inference; local embedding/ reranker	Operates without internet when model assets and packages staged locally	Model license, GPU compatibility, context and latency test
Hybrid governed	Local model for restricted workloads; external provider for approved classes	Gateway routes by data classification, health, cost and task	Verify downgrade and failure policy; no unsafe automatic exfiltration

Reference planning envelope	AI/RAG role	LLM role
Entry / API-led	8–16 vCPU, 32–64 GB RAM, 1–2 TB SSD; 1 backup location	External API; no GPU. Machine learning training sized separately.
Local midrange candidate	16–32 vCPU, 64–128 GB RAM, 2–4 TB SSD for API/RAG	Dedicated NVIDIA-compatible GPU host, 1 × ≥24 GB VRAM as a test starting point for selected quantized smaller model; final fit unproven.
Advanced candidate	2+ application roles, PostgreSQL HA, 4–10 TB initial knowledge/model storage	Dedicated multi-GPU host; for a BF16 70B-class model, weights alone ≈140 GB decimal; qualify ≥192 GB aggregate usable VRAM, topology and bandwidth.

این مقادیر «نقطه شروع آزمون» هستند، نه مشخصات قطعی خرید. فهرست مدل، دقت عددی، طول زمینه، نرخ توکن و تعداد کاربران، BOM نهایی را تعیین می کند. حافظه VRAM جمع شده بدون پشتیبانی مدل از تقسیم میان GPU ها قابل استفاده فرض نمی شود.

Sources: S01 · S03 · S04

Host and software inventory

برای نصب در سایت، نقش‌ها و محل داده‌ها باید مستقل از نام محصول مشخص شوند؛ هم‌میزبانی فقط پس از آزمون منابع و سیاست امنیتی انجام شود.

Role	Runtime and service	Persistent state	Approval gate
AI API	Release-matched ARG_AI server; OS/JDK/service wrapper from actual package	Accounts, chat, model registry, alerts, job state	Verify server build, DB, secrets and health API
Feature / ML worker	Isolated CPU or GPU jobs; queue semantics from selected implementation	Versioned features, model artifacts, experiments	Idempotent jobs and bounded retry
RAG ingestion	Parser/OCR, chunking, embedding and publisher workers	Original files, extracts, index manifest, ACL	Sanitization, provenance and failed-job quarantine
Search store	PostgreSQL + extension such as pgvector, or tested alternative	Tenant-scoped index, metadata, full text	Backup and permission-filter tests
LLM inference	vLLM or validated serving engine on Linux GPU node	Pinned model files, tokenizer, runtime manifest	Exact model compatibility and load benchmark
Provider gateway	Adapters for each protocol; secret manager integration	Policies, nonsecret usage ledger, redacted traces	Key isolation, quota and fallback tests
Operations	Proxy, PKI, metrics, logs, backup and alerting	Audit and restore evidence	RTO/RPO and offline test

Sources: C01 · S01 · S03 · S05 · S06

ML data pipeline and feature contracts

داده صنعتی باید پیش از آموزش و تحلیل به نمونه‌های قابل ردیابی تبدیل شود؛ هر ویژگی به تجهیز، نسخه نگاشت و بازه زمانی خود متصل است.

Stage	Exact contract	Acceptance
Extract	Approved historian projection; assetId, projectId, ENID/tagId, value, datatype, engineeringUnit, sourceTimestampUtc, quality	Source counts and quality totals reconcile
Normalize	Versioned unit conversion; range checks; missingness; downtime/maintenance masks	Known plant examples and conversion reversibility
Window	Feature interval, lookback, aggregation and event-time watermark; late-data rule	No future information in training labels
Label	Maintenance event, failure definition, operator validation and label provenance	Subject-matter sign-off on ground truth
Split	Temporal train/validation/test split; group by site/assets where leakage possible	Out-of-time holdout and per-unit metrics
Train	Dataset hash, feature version, code/container digest, random seed, model hyperparameters	Reproducible training manifest
Serve	Input schema and model version; abstain on stale/bad quality; bounded concurrency	Error budget, p95 latency and confidence calibration
Monitor	Drift, false alarm, data gaps, retrain trigger and rollback	Signed promotion and rollback exercise

Sources: [C01](#) · [C02](#) · [C03](#) · [S13](#)

ML commissioning and acceptance

این چکلیست جلوی نمایش یک نمودار آزمایشی به عنوان تحلیل واقعی سایت را می گیرد.

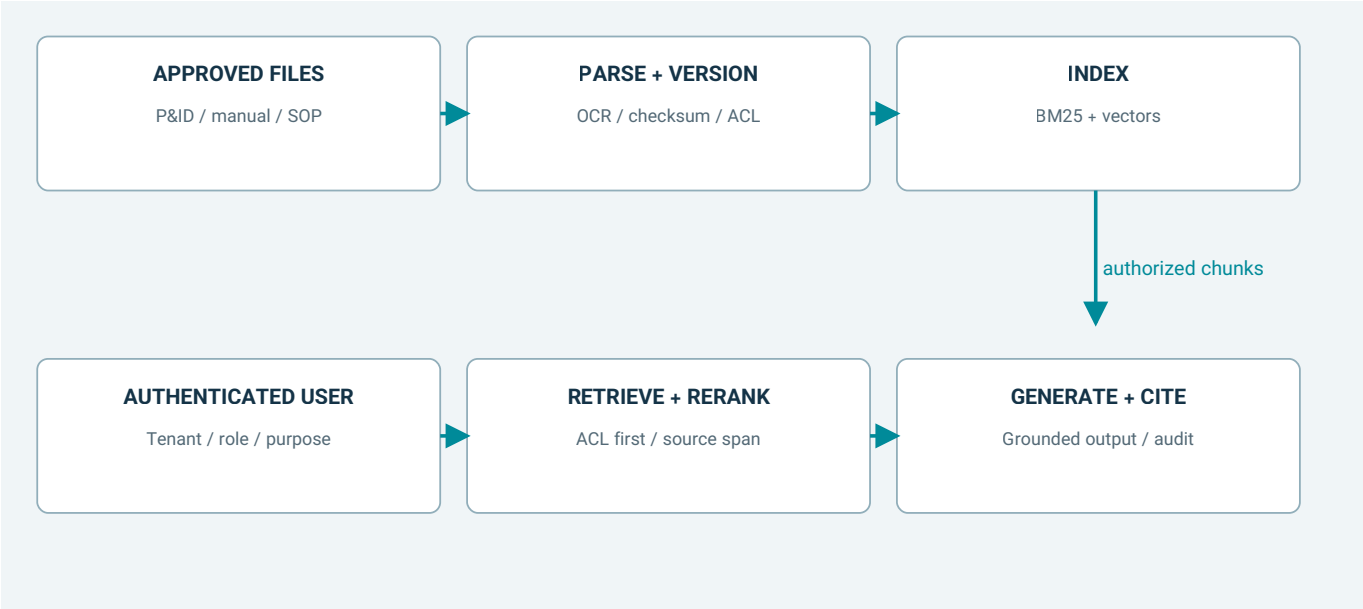
Gate	Implementation test	Evidence
M01 Input lineage	Replace generateSampleDataPoints and generateEnergyDataPoints in reviewed AI pages	A selected plant historian row traces to prediction and asset
M02 Baseline	Compare with seasonal naive, threshold rules or existing expert method	Accuracy / precision / recall by asset and operating regime
M03 Drift	Inject changed scale, missing values, late samples, failed sensor	Visible abstention and drift metric; no silent forecast
M04 Model lifecycle	Promote only approved model; pin artifact checksum and feature signature	Shadow test, rollback and compatibility record
M05 Concurrency	Run simultaneous users and training jobs against serving node	Training cannot starve live inference; bounded queue and timeouts
M06 Ethics/operations	Record purpose, source permissions and data retention	Operator can see limitations and model version

آستانه های عددی دقت و تأخیر باید با نوع مسئله و داده تاریخی واقعی سایت تعیین شوند. درصد دقت عمومی یا «پیش بینی خرابی» بدون تعریف رخداد و داده برچسب دار، معیار تحویل نیست.

Sources: C02 · C03 · S13

RAG ingestion and document governance

RAG برای دستورالعمل‌های بهره‌برداری، نقشه‌ها، دفترچه‌ها و سوابق استفاده می‌شود؛ نسخه و سطح محرمانگی هر سند باید در پاسخ حفظ شود.



Object	Mandatory metadata
Source	documentId, sourceURI, fileHash, documentVersion, documentType and revisionDate
Ownership	organizationId, projectId/siteId, owner, classification and permittedRoles
Evidence	page/section, char offsets or image/figure coordinates, extraction version
Index	parserVersion, chunkerVersion, embeddingModelId, dimension and indexRelease
Lifecycle	ingest timestamp, validFrom/validUntil, supersedesId and deletion/revocation

Sources: S05 · S12 · S13

RAG retrieval and authorization

فیلتر دسترسی باید قبل از ارسال قطعه به مدل اعمال شود؛ ذکر منبع پس از پاسخ، مجوز دسترسی را جایگزین نمی‌کند.

Step	Server-side requirement
Authenticate	Resolve user, tenant, site and document permissions from the authoritative identity service
Select	Filter eligible document versions, classification and validFrom/Until before vector/keyword search
Retrieve	Use full-text + vector candidate sets; verify ANN filtering returns enough tenant-authorized hits
Rerank	Reorder a small authorized set; enforce score floor and maximum context tokens
Generate	Treat documents as untrusted data; require exact page/section citations in each factual answer
Abstain	If evidence is absent or conflicting, return uncertainty and source references rather than invented plant instructions
Audit	Persist user, document versions, retrieved chunk IDs, model/prompt release, outcome and timestamps without raw secrets
Invalidate	On revocation or new revision, remove inaccessible chunks; reindex without cross-version mixing

در سناریوی صنعتی، پاسخ RAG نباید به صورت خودکار فرمان کنترلی صادر کند. متن یک سند ممکن است حاوی دستور تزریق شده باشد؛ ابزارها و مجوزها باید مستقل از متن بازیابی شده کنترل شوند.

Sources: S05 · S12 · S13

Vector and storage capacity example

این مثال، روش محاسبه را روشن می‌کند؛ ابعاد و حجم واقعی پس از انتخاب مدل embedding و نمونه‌برداری سند اندازه‌گیری می‌شوند.

Assumption	Example calculation
Corpus	100,000 approved chunks; 1,024 dimensions; 32-bit float representation
Raw vector bytes	$100,000 \times 1,024 \times 4 = 409,600,000$ bytes $\approx$ 0.41 GB decimal
Additional storage	HNSW graph, metadata, ACL, full-text index, original/PDF/scan files, OCR and replicas are extra
Versioning	A new embedding model may require a parallel index; reserve old and new during migration
Search benchmark	Test p95 retrieval latency at selectivity of project/role ACL and requested top-k
Backup	Logical metadata plus physical/index files and original documents restored at same revision

Do not infer total RAG disk from raw vectors alone. If filters and approximate indexes interact, run recall tests with actual per-project data; pgvector documents iterative scans as one mechanism for filtered ANN results.

Sources: [S05](#) · [S06](#)

Embedding models and Persian quality

مدل embedding و مدل تولید متن دو سرویس متفاوت هستند. انتخاب مدل باید با اسناد فارسی و انگلیسی واقعی ارگ آزمون شود.

Decision	Acceptance rule
Tokenizer/language	Evaluate Persian and English retrieval, transliterated tag IDs, acronyms and equipment labels
Model isolation	Use dedicated embedding endpoint/API key; chat provider may not implement /embeddings
Dimension	Persist model ID and vector dimension with each index; reject mixed-dimension inserts
Chunking	Keep headings, tables, figure caption, page and P&ID component associations
Multimodal input	OCR and image/diagram interpretation only when validated; rasterized schematic text is not authoritative by default
Retrieval eval	Measure recall@k, answer citation precision, unsupported claim rate and permission leakage
Update	Blue/green reindex on model change; compare retrieval before alias swap

در Main Backend فعلی کلاینت embedding با مسیر /embeddings و کلاینت گفت‌وگو با قرارداد جداگانه دیده می‌شوند. برای ARG_AI مستقل باید اتصال embedding طبق API واقعی سرور تأیید شود.

Sources: C04 · C05 · S05

Local LLM host and deployment sequence

استنتاج محلی یک سرویس جدا در شبکه خصوصی است. ابتدا سازگاری مدل و GPU آزمون می‌شود، سپس سیستم تولیدی از طریق درگاه ARG_AI به آن وصل می‌شود.

Step	Implementation and pass criterion
L01 Host	Approve rack, PSU, cooling, ECC RAM, GPU interconnect, NVMe and driver support; dedicated Linux GPU host
L02 Runtime	Install pinned NVIDIA driver and Container Toolkit where selected; capture nvidia-smi and signed image digest
L03 Model	Approve model license and checkpoint hash; stage tokenizer/config/weights offline; record quantization and context
L04 Serve	Start validated vLLM release on private address; choose model ID, context limit, concurrency, memory utilization
L05 Gateway	Put service authentication and TLS proxy in front; vLLM API key is not a complete protection for all routes
L06 Benchmark	Measure prefill, time-to-first-token, output tokens/s, p95/p99 latency, VRAM peak and OOM at target concurrency
L07 Quality	Compare approved plant prompts, Persian terminology and RAG citations; test provider outage and local-only mode
L08 Lifecycle	Pin model/service version; one active model promotion; retain prior version and rollback after test

Illustrative private endpoint (choose a validated model and release):

<https://llm.infra.plant.example/v1/chat/completions>

Never expose the raw inference server to OT or general users.

A local OpenAI-compatible endpoint is an interface choice, not proof of compatibility with ARG Main Backend Anthropic Messages calls.

Sources: [S01](#) · [S02](#) · [S03](#) · [S04](#) · [C04](#)

Provider adapters and API key lifecycle

کلید هر ارائه‌دهنده باید پشت سرور باقی بماند؛ مرورگر فقط با توکن کاربر ARG_AI صحبت می‌کند و هیچ کلیدی را دریافت نمی‌کند.

Adapter	Official interface shape	Required qualification
OpenAI	Bearer API key; provider-specific model and endpoint such as Responses	Pin endpoint/model, input/output schema, data controls, rate limits and request ID
Anthropic	Messages API with x-api-key and anthropic-version	System prompt/content blocks, streaming, tool schema, retries and token usage
Gemini	generateContent and x-goog-api-key	Request schema, safety, quota, retry and region/data rules
On-prem vLLM	OpenAI-style chat endpoint, private service authentication	Feature parity not assumed; check embeddings separately
Approved aggregator	Provider-specific compatibility may differ even when URL looks familiar	Contract tests per provider and model; avoid blanket API compatibility claim

Key control	Required practice
Provision	One server-side service identity/key per environment/provider; central vault reference in configuration
Rotate	Dual-key overlap if provider permits; staged cutover, smoke test, revoke and incident procedure
Restrict	Network egress allowlist, spend limits, rate caps, data classification and admin-only provider settings
Observe	Request ID, provider, token usage, cost bucket, error class and latency; redact prompts by policy

Sources: S07 · S08 · S09 · S10 · S11 · C01

Routing policy and failure behavior

انتخاب بین مدل محلی و API باید یک تصمیم قابل ردگیری بر پایه حساسیت داده و وضعیت سرویس باشد.

Data class	Default route	Fallback
Restricted OT / unreleased plant documents	Local model + local embeddings only	Queue or return unavailable; never auto-fallback to cloud
Internal approved	Local preferred or permitted external provider after explicit site policy	Use only approved provider with equivalent data controls
Public/open	Selected provider by measured quality, latency and budget	Second provider or local, within owner-approved limits

Example nonsecret routing policy (design, not current ARG_AI config):

restricted: local_only, external_fallback: false

internal: allowlist [local, approved_provider_A], log_payload: false

public: allowlist [approved_provider_A, local], budget_limit: site_policy

all: timeout_ms, retry_limit, max_tokens, per_user_rate_limit, audit_required

خطاهای ۴۲۹ و ۵XX نیازمند صف محدود، backoff و timeout هستند. تکرار خودکار بدون شناسه درخواست و محدودیت هزینه می‌تواند هزینه و پاسخ تکراری تولید کند.

Sources: S08 · S11 · S12

ARG integration and identity reconciliation

فرانت‌اند AI در کد فعلی نشست جداگانه دارد. یکپارچه‌سازی نهایی هویت، بدون مشاهده سورس سرور و آزمون نشست انجام‌شده فرض نمی‌شود.

Link	Reviewed state	Implementation gate
Browser → ARG_AI	Independent JWT from /api/v1/auth/login; token currently persisted in localStorage	Assess secure cookie/BFF or hardened storage; enforce TLS, expiry, refresh and origin policy
Frontend → ML	/api/v1/ml/analyze, /ml/train and /ml/health visible in client	Replace demo arrays with approved telemetry features
Frontend → RAG	No verified dedicated RAG route from reviewed standalone client	Implement versioned authorized endpoint or verify current server endpoint
Main Backend → AI	Embedded assistant uses Anthropic-style / messages; embedding client uses /embeddings	Choose whether to retain that path or route through provider adapters; test payload and auth
ARG_AI → plant data	No complete production data feed proven in reviewed UI	Signed interface contract for projectId, ENID, sourceTimestampUtc, quality, unit
AI output → UI	Analysis/forecast results should carry model and data version	Show stale, unavailable and evidence state; no implied process control

Sources: C01 · C02 · C03 · C04 · C05 · C06

Security and governance release gates

امنیت RAG و مدل محلی به اندازه شبکه و دیتابیس مهم است؛ متن اسناد، مدل یا درخواست کاربر حق افزایش سطح دسترسی ندارد.

Gate	Negative and recovery test
A01 Cross-tenant	A user from project B cannot retrieve chunks, embeddings or chat memory from project A
A02 Prompt injection	Malicious retrieved text cannot change system policy, call tools or reveal another document
A03 Secret handling	Provider keys absent from JS bundle, logs, traces, RAG context and downloadable config
A04 Egress	Restricted prompt never leaves site even when local model is down
A05 Document hygiene	Reject malicious files, detect OCR failure, scan uploads and preserve source hash
A06 AI actions	Assistant recommendation cannot directly dispatch write commands to DCS/PLC
A07 Supply chain	Pin model license/checkpoint, container image, dependencies and SBOM
A08 Privacy	Set provider retention and approved destinations per current contract; test deletion and data export

Sources: [S02](#) · [S08](#) · [S12](#) · [S13](#) · [S14](#)

Operations, backup and continuity

نسخه قابل بهره‌برداری باید با اختلال مدل، دیتابیس و ارائه‌دهنده رفتار روشن داشته باشد.

Metric or asset	Monitoring / recovery criterion
API and auth	Signed-in success rate, p95 latency, 401/403 and JWKS/session expiry if applicable
ML worker	Queue depth, failed jobs, training duration, feature age and model drift
RAG	Ingest failures, ACL rejection, search recall set, index version, result citation rate
LLM host	GPU memory/utilization, OOM, TTFT, output token rate, concurrent requests and throttles
External calls	Provider outages, 429, billed tokens, request IDs, budget consumption and data route
Backup set	AI application DB, prompts/policies, model checkpoints, original docs, index build manifest, keys via vault process
Restore	Recreate from pinned releases; restore DB/documents as one logical revision; rebuild index if chosen; smoke auth + retrieval
Availability	RTO/RPO negotiated per use case; HA needs service state and worker scheduling design, not only two VMs

Sources: S04 · S05 · S06 · S11

FAT/SAT acceptance matrix

این آزمون‌ها در محیط غیرتولیدی شروع می‌شوند و با نمونه تأییدشده سایت ادامه می‌یابند.

Test	Acceptance evidence
Real ML input	Choose 10 source tags across assets; trace historian row to feature, model result and UI. No generated sample array.
Time/quality	Bad-quality, late, timezone shift and missing samples show stale/abstain state.
RAG ground truth	At least 100 jointly reviewed questions across Persian/English and authorized roles; report retrieval recall@k, citation precision and unsupported claims.
Permission	Denied documents never appear in search candidates, model context, citation or conversational memory.
Local performance	Target sessions, prompt/context lengths and bursts; record p95/p99 end-to-end and GPU OOM=0 during soak.
Provider failure	Simulate 429, timeouts and provider outage; restricted data remains local and cost bound remains intact.
Reindex/restore	Document revocation, model rollback, vector rebuild and consistent restore demonstrated.
OT separation	Packet captures show only approved collector/DMZ flows; no AI-to-controller control path.
Commissioning sign-off	Plant IT, OT security, process owner, ARG ML and operations approve release manifest.

آستانه‌های پاسخ‌گویی، دقت و هزینه باید پیش از FAT در صورت جلسه تثبیت شوند. این جدول روش آزمون را مشخص می‌کند و جای نتیجه‌گیری از آزمونی که هنوز اجرا نشده را نمی‌گیرد.

Sources: C02 · C03 · S13 · S14

Execution sequence and deliverables

پیاده‌سازی مرحله‌ای است تا هر وابستگی قبل از خرید و نصب نهایی شواهد کافی داشته باشد.

Phase	Outputs and exit gate
0 Scope and audit	Current ARG_AI server JAR/source, OpenAPI, database schema, deployment script, model list; signed workload and data policy
1 Infrastructure	Hosts/VMs, DNS/PKI, approved storage, backed-up DB, GPU driver, private inference zone and vault references
2 Existing ML with real data	Feature contract, historian extraction, quality handling, baseline model, model registry and UI input replacement
3 Existing RAG integration	Document ownership, ingestion/OCR, chunk/embedding model, vector+keyword index, ACL and 100-question set
4 Model gateways	On-site model and/or provider adapters; policy, keys, budget, routing and secret rotation
5 Integrated FAT	Service compatibility, performance, security, restore and restricted data egress tests
6 Site SAT	Controlled data ramp, operator sign-off, observed load, asset mappings, training and handover
7 Operations	As-built ICD, port register, runbooks, backup/rollback, ownership, drift and cost dashboards

Stop point for final purchasing: complete phase 0 and benchmark the actual selected models. Reuse ARG_AI ML and RAG components after contract and security verification; provision only the supporting roles required by the actual release.

Sources: C01 · C04 · S01 · S13

Decision record for the ARG team

برای جلسه تصمیم‌گیری، خروجی‌های لازم از مالک AI، زیرساخت و بهره‌برداری در یک جدول آمده است.

Decision owner	Information to provide	Engineering consequence
ARG_AI owner	Current server release, source/POM, OpenAPI, database, model artifacts, training jobs and provider integration	Exact installation commands and migration plan
Plant automation	Approved historian export, tags, time/quality semantics and rate limit	Safe feature extraction and provenance
Plant security	Data classification, WAN allowance, proxy/Vault/PKI and logging policy	Local vs API vs hybrid route
Process engineer	Failure definitions, trusted manuals/P&IDs and permissible response use	ML targets, RAG evaluation and acceptance
IT infrastructure	GPU candidate, physical storage and failover target	Final qualified BOM and recovery
Commercial owner	Provider billing account, license and spending cap	Unit economics and procurement

اگر سرور مستقل ARG_AI اکنون RAG یا مدیریت مدل دارد، اول آن را با قرارداد واقعی و نسخه انتشار تطبیق می‌دهیم. سند حاضر معماری هدف و معیار را می‌دهد؛ برای دستور نصب محصول فعلی باید سورس یا بسته همان انتشار دیده شود.

Sources: C01 · C04 · C05

Source register and claim status

منابع کد به commit های بررسی شده و منابع بیرونی به ناشر اصلی اشاره دارند. محتوای نسخه های بیرونی ممکن است به روزرسانی شود؛ نسخه بسته اجرایی باید تثبیت گردد.

Claim class	Meaning
Source verified	Client/main-backend behaviors observed in linked code, not executed in a plant
Vendor documented	Capability or interface published by its maintainer; compatibility with ARG still requires tests
Engineering proposal	Topology, hardware starting envelopes, policies and FAT designed in this document
Not verified	ARG_AI ML and RAG are owner-confirmed; internal implementation and live site configuration were unavailable for code audit

C01 ARG standalone AI client: auth, ML, twin, providers
C02 ARG anomaly page sample input
C03 ARG prediction page sample input
C04 Main Backend agent model client: Anthropic Messages
C05 Main Backend embedding client: embeddings interface
C06 Main Backend agent action policy
C07 ARG frontend build and dependencies
S01 vLLM online serving and compatible endpoints
S02 vLLM API server authentication boundaries
S03 NVIDIA Container Toolkit installation
S04 NVIDIA Triton dynamic batching
S05 pgvector indexing, filtering, iterative scans
S06 PostgreSQL continuous archiving
S07 OpenAI API authentication and request IDs
S08 OpenAI API data controls by endpoint
S09 Anthropic Messages API
S10 Gemini API authentication
S11 Gemini API rate limits
S12 OWASP prompt injection
S13 NIST AI Risk Management Framework
S14 NIST SP 800-82 Rev. 3 OT security

Prepared 27 September 2026. No deployment, production access, model training or industrial load test was performed.